EAST SEARCH

2/25/06

L#	Hits	Search String	Databases
S1	1254	((integrated or digital) near2 circuit$1) with emulat$3	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S2	59	S1 and (distributed with (emulat$3 or processing))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S3	6387	circuit$1 with emulat$3	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S4	304	S3 and (distributed with (emulat$3 or processing))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S5	304	S2 or S4	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S6	47	S5 and (reconfigurable with (logic or interconnect$1))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S7	127	S5 and (circuit with element$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S8	16	S5 and (circuit with partition$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S9	7	S5 and (on-board with processing$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S10	38	S5 and (element$1 with state$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S11	67	S5 and ((monitor$3 or report$3 or test$3) with request$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S12	48	S5 and ((monitor$3 or report$3 or test$3) with command$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S13	27	S5 and (retriev$3 with state$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S14	93	S5 and ((analyz$3 or analysis) with data)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S15	84	S5 and ((detect$3 or report$3) with event$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S16	87	S5 and (board with circuit$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S17	10	S5 and (on-chip with processing)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S18	54	S5 and (local$2 with (monitor$3 or analyz$3 or analysis or report$3))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S19	30	S5 and (test$3 with (vector$1 or stimulus or stimulii))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S20	38	S5 and ((generat$3 or produc$3) with (vector$1 or stimulus or stimulii))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S21	3	S5 and (local$2 with (generat$3 or produc$3) with (vector$1 or stimulus or stimulii))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S22	25	S5 and ((emulat$3 near2 system) with board$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S23	89	S5 and (workstation)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S24	8	S5 and (EDA with software)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S25	167	S6 or S8 or S9 or S10 or S11 or S12 or S13 or S17 or S18 or S19 or S20 or S21 or S22 or S2	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S26	210	S7 or S14 or S15 or S16 or S23	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S27	133	S25 and S26	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S28	167	S25 or S27	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S29	1254	((integrated or digital) near2 circuit$1) with emulat$3	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S30	59	S29 and (distributed with (emulat$3 or processing))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S31	6387	circuit$1 with emulat$3	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S32	304	S31 and (distributed with (emulat$3 or processing))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S33	304	S30 or S32	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S34	47	S33 and (reconfigurable with (logic or interconnect$1))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S35	127	S33 and (circuit with element$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S36	16	S33 and (circuit with partition$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB

S37	7	S33 and (on-board with processing$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S38	38	S33 and (element$1 with state$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S39	67	S33 and ((monitor$3 or report$3 or test$3) with request$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S40	48	S33 and ((monitor$3 or report$3 or test$3) with command$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S41	27	S33 and (retriev$3 with state$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S42	93	S33 and ((analyz$3 or analysis) with data)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S43	84	S33 and ((detect$3 or report$3) with event$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S44	87	S33 and (board with circuit$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S46	10	S33 and (on-chip with processing)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S48	54	S33 and (local$2 with (monitor$3 or analyz$3 or analysis or report$3))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S49	30	S33 and (test$3 with (vector$1 or stimulus or stimulii))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S51	38	S33 and ((generat$3 or produc$3) with (vector$1 or stimulus or stimulii))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S52	3	S33 and (local$2 with (generat$3 or produc$3) with (vector$1 or stimulus or stimulii))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S53	25	S33 and ((emulat$3 near2 system) with board$1)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S54	89	S33 and (workstation)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S55	8	S33 and (EDA with software)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S56	167	S34 or S36 or S37 or S38 or S39 or S40 or S41 or S45 or S46 or S47 or S48 or S49 or S50 o	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S57	210	S35 or S42 or S43 or S44 or S51	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S58	133	S53 and S54	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S59	167	S53 or S55	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S60	4	6,265,894.pn. or "5,777,489".pn.	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S61	1318	((integrated or digital) near2 circuit$1) with emulat$3	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S62	62	S58 and (distributed with (emulat$3 or processing))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S63	62	S58 and (distributed with (emulat$3 or processing))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S64	6696	circuit$1 with emulat$3	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S67	321	S61 and (distributed with (emulat$3 or processing))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S69	321	S60 or S62	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S70	50	S63 and (reconfigurable with (logic or interconnect$1))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S71	10	S63 and (on-chip with processing)	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S72	32	S63 and (test$3 with (vector$1 or stimulus or stimulii))	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S73	40	**S63 and ((generat$3 or produc$3) with (vector$1 or stimulus or stimulii))**	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S74	3	S63 and ((local$2 or on-chip) with (generat$3 or produc$3) with (vector$1 or stimulus or stimuli	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB
S75	53	S65 or S66 or S67 or S68	US-PGPUB; USPAT; EPO; JPO; DERWENT; IBM_TDB

10/003184 **Frederic Reblewski**

EAST SEARCH

2/25/06

Results of search set S115

Document	Kind Codes	Title	Issue Date	Current OR	Abstract
US 20050071716	A1	Testing of reconfigurable logic and interconnect sources	20050331	714/725	

US 20040260530 A1	Distributed configuration of integrated circuits in an emulation system	20041223	703/23
US 20040220891 A1	Neural networks decoder	20041104	706/12
US 20040181497 A1	Neural networks	20040916	706/23
US 20040078187 A1	Emulation components and system including distributed routing and configuration of emulation	20040422	703/28
US 20040059876 A1	Real time emulation of coherence directories using global sparse directories	20040325	711/141
US 20040044514 A1	Polymorphic computational system and method in signals intelligence analysis	20040304	703/23
US 20040034841 A1	Emulation components and system including distributed event monitoring, and testing of an IC	20040219	716/8
US 20030233504 A1	Method for detecting bus contention from RTL description	20031218	710/107
US 20030149675 A1	Processing device with intuitive learning capability	20030807	706/2
US 20030105617 A1	Hardware acceleration system for logic simulation	20030605	703/14
US 20030074178 A1	Emulation system with time-multiplexed interconnect	20030417	703/25
US 20020177990 A1	Distributed logic analyzer for use in a hardware logic emulation system	20021128	703/28
US 20020161568 A1	Memory circuit for use in hardware emulation system	20021031	703/25
US 20020116168 A1	METHOD AND SYSTEM FOR DESIGN VERIFICATION OF ELECTRONIC CIRCUITS	20020822	703/28
US 20020066065 A1	Method, apparatus, and program for multiple clock domain partitioning through retiming	20020530	716/6
US 6922664 B1	Method and apparatus for multi-sensor processing	20050726	703/13
US 6920416 B1	Electronic systems testing employing embedded serial scan generator	20050719	703/13
US 6832178 B1	Method and apparatus for multi-sensor processing	20041214	702/189
US 6732068 B2	Memory circuit for use in hardware emulation system	20040504	703/24
US 6694464 B1	Method and apparatus for dynamically testing electrical interconnect	20040217	714/725
US 6684318 B2	Intermediate-grain reconfigurable processing device	20040127	712/15
US 6571370 B2	Method and system for design verification of electronic circuits	20030527	716/4
US 6567962 B2	Method, apparatus, and program for multiple clock domain partitioning through retiming	20030520	716/6
US 6496918 B1	Intermediate-grain reconfigurable processing device	20021217	712/15
US 6415188 B1	Method and apparatus for multi-sensor processing	20020702	700/67
US 6377912 B1	Emulation system with time-multiplexed interconnect	20020423	703/28
US RE37488 E	Heuristic processor	20011225	706/14
US 6266760 B1	Intermediate-grain reconfigurable processing device	20010724	712/15
US 6052524 A	System and method for simulation of integrated hardware and software components	20000418	703/22
US 5960191 A	Emulation system with time-multiplexed interconnect	19990928	703/28
US 5956518 A	Intermediate-grain reconfigurable processing device	19990921	712/15
US 5943490 A	Distributed logic analyzer for use in a hardware logic emulation system	19990824	703/28
US 5940603 A	Method and apparatus for emulating multi-ported memory circuits	19990817	716/5
US 5937154 A	Manufacturing functional testing of computing devices using microprogram based functional tes	19990810	714/30
US 5841670 A	Emulation devices, systems and methods with distributed control of clock domains	19981124	703/23
US 5838948 A	System and method for simulation of computer systems combining hardware and software inte	19981117	703/27
US 5761077 A	Graph partitioning engine based on programmable gate arrays	19980602	716/7
US 5684721 A	Electronic systems and emulation and testing devices, cables, systems and methods	19971104	703/23
US 5663900 A	Electronic simulation and emulation system	19970902	
US 5621651 A	Emulation devices, systems and methods with distributed control of test interfaces in clock don	19970415	703/23
US 5517597 A	Convolutional expert neural system (ConExNS)	19960514	706/26
US 5475793 A	Heuristic digital processor using non-linear transformation	19951212	706/14
US 5452239 A	Method of removing gated clocks from the clock nets of a netlist for timing sensitive implement	19950919	703/19

US 5377306 A	Heuristic processor	19941227	706/14
US 5357597 A	Convolutional expert neural system (ConExNS)	19941018	706/25
US 5222193 A	Training system for neural networks and the like	19930622	706/25
US 5113500 A	Multiple cooperating and concurrently operating processors using individually dedicated memor	19920512	710/305
US 5087826 A	Multi-layer neural network employing multiplexed output neurons	19920211	706/38
US 4961002 A	Synapse cell employing dual gate transistor structure	19901002	365/185.03
US 4896053 A	Solitary wave circuit for neural network emulation	19900123	706/38
US 4802103 A	Brain learning and recognition emulation circuitry and method of recognizing events	19890131	706/38
US 4773024 A	Brain emulation circuit with reduced confusion	19880920	706/20

Results for "(('on-chip processing' and programmable)<in>metadata)"
Your search matched 4 of 1320520 documents.
A maximum of 100 results are displayed, 25 to a page, sorted by Relevance in Descending order.

e-mail printer

» Search Options

View Session History

New Search

Modify Search

(('on-chip processing' and programmable)<in>metadata) Search

Check to search only within this results set

Display Format: Citation Citation & Abstract

» Key

IEEE JNL	IEEE Journal or Magazine
IEE JNL	IEE Journal or Magazine
IEEE CNF	IEEE Conference Proceeding
IEE CNF	IEE Conference Proceeding
IEEE STD	IEEE Standard

view selected items Select All Deselect All

1. **Programmable multi-task on-chip processing for CMOS imagers**
Boussaid, F.; BermakA; Bouzerdoum, A.;
MEMS, NANO and Smart Systems, 2003. Proceedings. International Conference on
20-23 July 2003 Page(s):227 - 232
AbstractPlus | Full Text: PDF(630 KB) IEEE CNF
Rights and Permissions

2. **A 256×256 pixel smart CMOS image sensor for line-based stereo vision applicatior**
Ni, Y.; Guan, J.;
Solid-State Circuits, IEEE Journal of
Volume 35, Issue 7, July 2000 Page(s):1055 - 1061
Digital Object Identifier 10.1109/4.848217
AbstractPlus | References | Full Text: PDF(244 KB) IEEE JNL
Rights and Permissions

3. **A 256×256-pixel smart CMOS image sensor for line based stereo vision applicatio**
Yang Ni; Guan, J.H.;
Solid-State Circuits Conference, 1999. ESSCIRC '99. Proceedings of the 25th European
21-23 Sept. 1999 Page(s):258 - 261
AbstractPlus | Full Text: PDF(408 KB) IEEE CNF
Rights and Permissions

4. **Shunting inhibition-based on-chip processing for CMOS imagers**
Boussaid, F.; Bermak, A.; Bouzerdoum, A.;
Neural Information Processing, 2002. ICONIP '02. Proceedings of the 9th International Conference on
Volume 3, 18-22 Nov. 2002 Page(s):1310 - 1314 vol.3
AbstractPlus | Full Text: PDF(374 KB) IEEE CNF
Rights and Permissions

Terms used **on chip processing programmable** Found **8** of **171,143**

Sort results by: publication date

Display results: expanded form

Save results to a Binder

Search Tips

Open results in a new window

Try an Advanced Search

Try this search in The ACM Guide

Results 1 - 8 of 8

Relevance scale

1 A scalable, clustered SMT processor for digital signal processing

Mladen Berekovic, Sören Moch, Peter Pirsch

June 2004 **ACM SIGARCH Computer Architecture News**, Volume 32 Issue 3

Publisher: ACM Press

Full text available: pdf(356.32 KB) Additional Information: full citation, abstract, references

A scalable, distributed, processor architecture is presented that emphasizes on high performance computing for digital signal processing applications by combining high frequency design techniques with a very high degree of parallel processing on a chip. The architecture is based on a superscalar processor model with a modified Tomasulo scheme [1], that was extended to eliminate all central control structures for the data flow and to support simultaneous instruction issue from multiple independen ...

2 CAD: Design and optimization of MOS current mode logic for parameter variations

Hassan Hassan, Mohab Anis, Mohamed Elmasry

April 2004 **Proceedings of the 14th ACM Great Lakes symposium on VLSI**

Publisher: ACM Press

Full text available: pdf(304.83 KB) Additional Information: full citation, abstract, references, index terms

An automated optimization-based design strategy is proposed for single-level MOS Current Mode Logic (MCML) gates to overcome the complexities of the gate design procedure. The proposed design methodology determines the values of the design variables that achieve the minimum power dissipation point while attaining the required performance. The proposed design methodology has the advantage of speed, accuracy, and ability to include a large number of parameters in the design problem. Moreover, a fo ...

Keywords: MCML, automation, design, optimization, technology scaling, variation

3 Hardware/Software Co-testing of Embedded Memories in Complex SOCs

Bai Hong Fang, Qiang Xu, Nicola Nicolici

November 2003 **Proceedings of the 2003 IEEE/ACM international conference on Computer-aided design**

Publisher: IEEE Computer Society

Full text available: pdf(145.29 KB) Additional Information: full citation, abstract, index terms

A novel approach for testing embedded memories in complexsystems-on-a-chip (SOCs) is presented. The proposedsolution aims to balance the usage of the existing on-chipresources and dedicated design for test (DFT) hardwaresuch that the functional power constraints are not exceededdduring test while trading-off the testing time againstDFT area

and performance overhead. The suitability ofsoftware-centric and hardware-centric approaches for embeddedmemory testing is examined and to combine the advanta ...

4 Design space exploration and architectural design of HW/SW systems: Hardware support for real-time embedded multiprocessor system-on-a-chip memory management

Mohamed Shalan, Vincent J. Mooney

May 2002 **Proceedings of the tenth international symposium on Hardware/software codesign**

Publisher: ACM Press

Full text available: pdf(533.74 KB) Additional Information: full citation, abstract, references, index terms, review

The aggressive evolution of the semiconductor industry --- smaller process geometries, higher densities, and greater chip complexity --- has provided design engineers the means to create complex high-performance Systems-on-a-Chip (SoC) designs. Such SoC designs typically have more than one processor and huge memory, all on the same chip. Dealing with the global on- chip memory allocation/de-allocation in a dynamic yet deterministic way is an important issue for the upcoming billion transistor mu ...

Keywords: Atalanta, SoCDMMU, System-on-a-Chip, dynamic memory management, embedded systems, real-time operating systems., real-time systems, two-level memory management

5 A reconfigurable multi-function computing cache architecture

Hue-Sung Kim, Arun K. Somani, Akhilesh Tyagi

February 2000 **Proceedings of the 2000 ACM/SIGDA eighth international symposium on Field programmable gate arrays**

Publisher: ACM Press

Full text available: pdf(992.08 KB) Additional Information: full citation, abstract, references, citings, index terms

A considerable portion of a chip is dedicated to a cache memory in a modern microprocessor chip. However, some applications may not actively need all the cache storage, especially the computing bandwidth limited applications. Instead, such applications may be able to use some additional computing resources. If the unused portion of the cache could serve these computation needs, the on-chip resources would be utilized more efficiently. This presents an opportunity to explore the reconfigurat ...

6 Microservers: a new memory semantics for massively parallel computing

Jay B. Brockman, Peter M. Kogge, Thomas L. Sterling, Vincent W. Freeh, Shannon K. Kuntz

May 1999 **Proceedings of the 13th international conference on Supercomputing**

Publisher: ACM Press

Full text available: pdf(1.40 MB) Additional Information: full citation, references, citings, index terms

Keywords: massively parallel, microserver, processing-in-memory

7 Memory bandwidth limitations of future microprocessors

Doug Burger, James R. Goodman, Alain Kägi

May 1996 **ACM SIGARCH Computer Architecture News , Proceedings of the 23rd annual international symposium on Computer architecture ISCA '96**, Volume 24 Issue 2

Publisher: ACM Press

Full text available: pdf(1.60 MB) Additional Information: full citation, abstract, references, citings, index terms

This paper makes the case that pin bandwidth will be a critical consideration for future microprocessors. We show that many of the techniques used to tolerate growing memory latencies do so at the expense of increased bandwidth requirements. Using a decomposition of execution time, we show that for modern processors that employ aggressive memory latency tolerance techniques, wasted cycles due to insufficient bandwidth generally exceed those due to raw memory latencies. Given the importance of ma ...

8 A laboratory for teaching parallel computing on parallel structures

Lan Jin, Lan Yang

March 1995 **ACM SIGCSE Bulletin , Proceedings of the twenty-sixth SIGCSE technical symposium on Computer science education SIGCSE '95**, Volume 27 Issue 1

Publisher: ACM Press

Full text available: pdf(541.68 KB) Additional Information: full citation, abstract, references, citings, index terms

For the effective use of a laboratory for teaching parallel processing, it is desirable to have parallel systems that can implement various parallel structures at hardware or software level. Such systems developed in our laboratories are described in this paper. They are a multi-computer with reconfiguration and the PVM (Parallel Virtual Machine) with structural implementation. The paper proposes a methodology and several classes of problems for teaching message-passing programming on paral ...

Results 1 - 8 of 8